Transformation theory for finite elements

Robert C. Kirby*

Despite the immense computataional power afforded by modern computers, enabling advanced algorithms such as the finite element method for partial differential equations is inhibited by implementation difficulties. Given the wide range of different variational forms and approximating spaces proposed in the literature, it can be useful to provide practitioners with flexibility in what methods they implement, even enabling theoretically sound methods that are infrequently applied because of code complexity.

Among many projects seeking to automate portions of finite element methods, the FIAT project provides both a theoretical framework and code infrastructure for constructing and evaluating very general finite element polynomial bases on a reference element. The perspective is flexible enough to allow not only arbitrary order Lagrange and Hermite elements, but also those of Argyris, Morley, Raviart-Thomas, Nedelec, Arnold-Winther, and so on. Codes for finite elements such as the FEniCS Form Compiler (FFC), PyLith, and PETSc use FIAT-generated basis functions within their finite element calculations.

While these codes interface FIAT, they currently only use a subset of elements that FIAT can construct. Beyond evaluating reference element basis functions, one must also understand how these bases maps from the reference element to each element of the mesh. Standard Lagrange elements transform nicely under affine maps – on each element of a mesh, the nodal basis functions are simply the image of a particular nodal basis function on the reference element. However, most other finite elements do not share this property. Generally, one must first map the reference element basis functions to a given element and then apply a linear transformation to obtain the nodal basis functions on that element.

It is our goal to establish the structure of this transformation for as wide a class of finite elements as possible. It turns out that classic ideas from finite element theory such as interpolation equivalence give information about this transformation, indicating the complexity of applying it. Namely, we show for typical interpolation-equivalent elements that the matrix must be block-diagonal, or equivalently that the graph of the matrix is reducible. For elements that are not interpolation-equivalent, one must proceed with a two-stage transformation using rectangular matrices, each of which also has particular discrete structure.

*Department of Mathematics and Statistics, Texas Tech University, Lubbock, TX 79409-1042. Research supported by the United States Department of Energy Office of Science under project number DE-FG02-07ER25821.